TITLE OF THE INVENTION

QUERIES-AND-RESPONSES PROCESSING METHOD,
QUERIES-AND-RESPONSES PROCESSING PROGRAM,
QUERIES-AND-RESPONSES PROCESSING PROGRAM RECORDING
MEDIUM, AND QUERIES-AND-RESPONSES PROCESSING APPARATUS

BACKGROUND OF THE INVENTION

Field of the Invention

[0001] The present invention relates to a query-and-response processing method for querying a database in a natural language as to information about a person or thing such as an organization contained in the database and presenting the information which a user requires.

Description of the Prior Art

[0002] Many systems that receive queries about data in a database and present a corresponding entry or data provided by processing the entry have been prototyped in laboratories and described in many papers .

[0003] However, the systems conventionally studied accept queries like "How high is the Eiffel Tower?" or "What nationality is the winner of the giant slalom in the Lillehammer Winter Olympic Games?" that are unambiguous and would yield a small number of answers.

[0004] On the other hand, text-based searches will increase that are ambiguous and sometimes yield a huge number of results that are difficult to determine their appropriateness. For example, if technologies relating to a particular technology are searched for among a huge number of patent documents, it is impossible to provide a clear single answer to the query. Therefore technologies used in the conventional query-response systems cannot be used readily in cases where a query may yield various answers.

[0005] As in the example in which patent documents relating to a technology are searched for, a text-based search in practice is not a simple query-and-response session, data to be retrieved is ambiguous, and yield a huge number of results that are difficult to determine their appropriateness. Therefore if unprocessed search results are presented to a user in a conventional system, the user has to perform a number of tasks such as the selection and extraction of required information.

SUMMARY OF THE INVENTION

[0006] In view of these problems, it is an object of the present invention to analyze the intention of a query provided by a user to reduce search result information to an amount manageable for the user, sort out the result information, and present it in an easily readable form to the user by selecting information.

[0007] To achieve the object, the present invention provides a query-and-response processing method for receiving a search request involving a query input by a user and searching a database to present search results to the user, comprising the steps of analyzing the search request provided by the user, generating search criteria based on the result of the search request analysis, and searching the database according to the search criteria. The main characteristics of the present invention also include the steps of analyzing the intention of the query based on the result of the search request analysis, selecting items to be presented to the user from the search results, determining an output form according to the result of the analysis of the intention of the query, and adjusting the number and output form of the search results according to the needs of the user to present the search results to the user.

[0008] The intention of the query provided by the user may be determined by finding a key topic of the search request and selecting main items concerning the key topic and items relating to the main items as items to be presented to process the search results.

[0009] The process described above may be implemented by a computer and a software program. The program may be stored on an appropriate recording medium such as a portable-medium memory, semiconductor memory, and hard disk that are readable by the computer.

BRIEF DESCRIPTION OF THE DRAWINGS

[0010] The present invention will be described below in detail with reference to the accompanying drawings. However, the present invention is not limited to embodiments described below.

[0011] FIG. 1 shows an exemplary configuration of a query-and-response processing apparatus in which the present invention is implemented;

FIG. 2 shows an example of a database to be searched;

FIG. 3 shows an example of a search request analysis;

FIG. 4 shows an example of an analysis dictionary used by a search request analyzer;

FIG. 5 shows a process flow in a search criteria generator;

FIG. 6 is a diagram for illustrating a process performed by a query intention analyzer and an output formatter;

FIG. 7 shows the flow of a process performed by the query intention analyzer;

FIG. 8 shows the flow of a process performed by the output formatter;

FIG. 9 is a diagram for illustrating the replacement of a word in the search request in the output formatter;

FIG. 10 is a diagram for illustrating an example of an embodiment of the search criteria generator;

FIG. 11 shows an example of another embodiment of the search criteria generator;

FIG. 12 shows an example of another embodiment of the search criteria generator;

FIG. 13A illustrates an example of search criteria widening;

FIG. 13B illustrates an example of search criteria widening;

FIG. 14A shows different examples of the method for widening search criteria;

FIG. 14B shows different examples of the method for widening search criteria;

FIG. 14C shows different examples of the method for widening search criteria;

FIG. 15 shows an example of a process for confirming a search area;

FIG. 16 shows an example of the determination of the search area;

FIG. 17 shows an example of another embodiment of the search criteria generator;

FIG. 18 shows an example of another embodiment of the search generator and a search executor;

FIG. 19 shows an example of a text complement process when text is entered into the database;

FIG. 20 shows the flow of a process for entering the text into the database; and

FIG. 21 shows an example of a process for using a secondary database during a search.

DETAILED DESCRIPTION OF PREFERRED EMBODIMENTS

[0012] FIG. 1 shows an exemplary configuration of a query-and-response processing apparatus in which the present invention is implemented. The query-and-response processing apparatus 1 comprises a CPU, memory, and a software program for implementing the following means.

[0013] A search request by a user is input through a search request input module 11. In an embodiment, the search request is input by the user in the form of a query in a natural language.

The search request may be any types of input such as a keyboard input, voice input through a microphone, an input from an information terminal connected to the apparatus through a communications line, or an input from an application program.

[0014] A search request analyzer 12, which is a search requesting module, analyzes the structure of the search request input through the search request input module 11 to extract things acting as the topic of the query and constrains on the topic. A search criteria generator 13, which is a search criteria generating module, receives an output from the search request analyzer 12 and translates it into search criteria for finding entries in a database 15. A search executor 14, which is a search executing module, searches through the database 15 based on the provided search criteria and returns search results.

[0015] A query intention analyzer 16, which is a query intention analyzing module, determines the intention of the query from the user based on the result of the search request analysis and determines a presenting way in which the information is presented to the user. An output formatter 17, which is an output formatting module, receives the output from the query intention analyzer 16, extracts information from the search results, determines items to be presented, adjusts their presentation format, and presents the search result in a form easily readable to the user.

[0016] The embodiment will be described below mainly with respect to an example in which a tabular database 15 as shown in FIG. 2 is searched through, for simplicity. Important items in the database 15 are marked with a flag ("*" in FIG. 2) indicating that they are key items. The database 15 searched is not necessarily limited to the tabular database as shown in FIG. 2. It may be one that consists of tagged structured text data.

[0017] For example, the tabular database 15 shown in FIG. 2 may be readily changed into the following tagged structured text data:

[0018] <organization information> <company information> <company name> company A </company name> <location> Kawasaki </location> <representative> Yamamoto </representative> <product information> <product class> keyboard </product class> … </company information> </organization information>

[0019] For convenience, upper level items such as "company information" and "product information" shown in FIG. 2 or semantic information corresponding to them are sometimes called category hereinafter.

[0020] FIG. 3 shows an example of a search request analysis. When a search request 20 provided by a user in a natural language is input to the search request analyzer 12, the search request analyzer 12 analyzes its sentence structure through a known natural language analysis (such as a morpheme analysis). Information about components of the input sentence and the relationship among them are yielded as the search results 21. The search criteria generator 13 uses the search results 21 to generate a search command in which criteria like search criteria 22 are specified. In this example, the information to be searched is "organization information" and a search for an entry having the product class, "keyboard", is requested. The search command with the search criteria 22 is provided to the search executor 14 and the search executor 14 searches through the database 15.

[0021] FIG. 4 shows an example of an analysis dictionary 23 used by the search request analyzer 12. The analysis dictionary 23 for a natural language contains parts of speech, semantic information, and other information associated with respective word spellings. Natural language analysis technologies using such an analysis dictionary 23 are well known and therefore the detailed description of which will be omitted herein.

[0022] FIG. 5 shows a process flow in the search criteria generator 13. The search criteria generator 13 inputs the result analyzed by the search request analyzer (step S1), and determines additional search criteria (step S2). A predicate (verb) may be extracted as search criteria to make an additional search criteria, or components other than a search focus may be extracted to add the search criteria. Then criteria are generated according to each component extracted (step S3).

[0023] FIG. 6 illustrates a process performed by the query intention analyzer 16 and the output formatter 17. The query intention analyzer 16 analyzes a query to determine the topic of the query. In this example, that a query is "company" of topic 25 is output as a result of the analysis.

[0024] Then, the output formatter 17 receives the search results from the search executor 14 and the output (the query topic 25 in this example) from the query intention analyzer

16, selects items to be presented to the user, and determines in which format they should be presented. In this example, it is determined that "company name", "location", "representative", and "product class" of the item are presented in this order as the output 26 from the output formatter 17.

[0025] FIG. 7 shows the flow of a process performed by the query intention analyzer 16. First, the analysis result for the search request is input into the query intention analyzer 16 from the search request analyzer 12 (step S10). Then a phrase indicating a request is excluded from the analysis result (step S11). The phrase indicating a request is "want to find" in "want to find companies selling keyboards", for example.

[0026] Then it is determined whether the top level component is a noun phrase (step S12). If it is a noun phrase, the core noun (typically the last noun) is extracted (step S13).

[0027] If the top level component is not a noun phrase, it is determined whether it is a verb phrase (step S14). If the top level component is a verb phrase, that is, a predicate, a subject with a postpositional word such as "は (ha)"or " が (ga)" indicative of a subject in a Japanese sentence is extracted (step S15). Finally, the results extracted at steps S13 and S15 are provided to the output formatter 17 as the query topic (step S16).

[0028] If no query topic is found, the output formatter 17 is informed of it and outputs search results in a conventional manner without formatting the output according to the present invention.

[0029] FIG. 8 shows the flow of process performed by the output formatter 17. The output formatter 17 receives the query topic from the query intention analyzer 16 (step S20) and selects items to be presented as follows (step S21). If an item in the database 15 matches the query topic, all items belonging to the topic item are presented. In particular, if the query topic 25 is "company" as shown in FIG. 6 in a search through the database 15 shown in FIG. 2, all the items (company name, location, and representative) of the company information are selected as items to be presented. Concerning to related items other than the topic item (company information), for example, the product information, "product class" of a main item is selected as an item to be presented. As a result, the company name, location, representative, product class, … are selected as items (output items) to be presented, as shown in FIG. 6. The selection of the items to be presented is shown only as an example, so other selection methods may also be used. For example, information about the intention of the query analyzed by the query intention analyzer 16 is used to select items based on selection criteria predefined by the user.

[0030] The output formatter 17 then sorts the selected items to be presented (step S22). The topic item and related items to be presented are sorted in this order for the sort of the presented items. If the name of an item in the database 15 is different from a corresponding word extracted from the search request and the search criteria generator 13 replaces the word in the search request with the item name to generate search criteria, the item name in the presentation items is re-replaced with the word in the search request (step S23). A specific example of this replacement will be described later.

[0031] A presentation item for which a value is specified as search criteria is excluded from the selected presentation items (step S24). This is performed because the value of that item is known to the user and therefore there is no need to present the item. Instead of presenting the known value, it is added to the output information as the description of the other presentation items in their entirety.

[0032] If the search results in the database 15 do not include search data for which a value is specified, the item is excluded from the presentation items (step S25). This is because it is useless to present the item for which there is no data having the specified value. The result of the output formatting is provided to a presentation module 18 and the presentation module 18 presents it as the final results to the user.

[0033] Examples of a query topic and output order will be illustrated below. If search request Q1 is a query, "Which companies sell keyboards?", the query topic would be "company name" because the top level component in the result of the search request analysis is "company". Thus, presentation items, "company name", "location", "representative", "product class", … are presented in this order.

[0034] On the other hand, if search request Q2 states "Want to know about keyboards currently available", the query topic extracted by the query intention analyzer 16 would be "keyboard". Thus, the output presentation items, "product class", "product name", "price", "company name", … are presented in this order.

[0035] The search criteria of the search request Q1 are "product class = keyboard", which is the same as that of search request Q2. However the query topics of these requests are different from each other. Therefore different items are selected as search results for presentation.

[0036] An example of sorting based on topic items will be provided below. It is assumed that the results of a search performed by the search executor 14 are as follows:

[Search results]

Company name	Product class
Company B Japan	Enhanced electronic keyboard for general use
Company A U.S.A.	Transparent keyboard
Company B Japan	Sampling keyboard
Musical instrument company C	Electronic keyboard for PC music
Company B Japan	Low-price electronic keyboard with capabilities of high-end product

[0037] If the topic is "company name", the following sorting would result.

[Sorting result]

Company name	Product class
Company A U.S.A.	Transparent keyboard
Company B Japan	Enhanced electronic keyboard for general use
Company B Japan	Sampling keyboard
Company B Japan	Low-price electronic keyboard with capabilities of high-end product

Musical instrument company C	Electronic keyboard for PC music

[0038] As described above, the output formatter 17 receives a topic item output from the query intention analyzer 16 and sorts a set of searched items based on the topic item. This enables data items having the same topic item value to be arranged adjacently to each other, resulting in a readable presentation.

[0039] FIG. 9 illustrates the replacement of words in a search request in the output formatter 17 (step S23 in FIG. 8). For example, it is assumed that a query, "Which makers are developing PCs for children?", is provided as a search request 20. The search request analyzer 12 produces analysis results 21. If the database 15 does not contain the item name, "maker", and instead contains an item, "organization", search criteria 22 in which the word "maker" is replaced with the item name "organization" contained in the database 15 are generated by referencing a thesaurus or a word replacement dictionary (not shown). The output formatter 17 replaces the item name "organization" in search result presentation items with the original word "maker" in the search request. This enables a presentation which the user may read easily and insensibly of the items.

[0040] As described above, if a user uses words such as "maker" and "personal computer" in a query representing a search request and the database uses different corresponding item names such as "organization" and "product class", item names in the database 15 are replaced with the words used in the user's search request when the information is presented by item, thereby improving the readability of the information.

[0041] An example of a process performed by the output formatter 17 at step S24 in FIG. 8 will be described below. Assumption is that a search request is "Want to know about digital cameras from Company A", for example. The search criteria would be "SELECT product information WHERE product class = digital camera company name = Company A", for example.

[0042] The output formatter 17 selects items to be presented from the results of this search. "Product class, product name, price and company name" are selected during the first presentation item selection, because the query topic is "digital camera". But, "product class = digital camera" and "company name = Company A" are items having the values specified as search criteria, and these values are common to all the search results, so the presentation of this information is unnecessary. Therefore, these items having the values specified as the search criteria are excluded from presentation items during the second presentation item selection, thereby yielding "product name, price" as presentation items. The neglected values in "product class" and "company name" are collectively provided at the top of the list of the rest of the presentation items as description that qualify the items, if necessary. This avoids displaying redundant items and the information for which a value is specified can be displayed as the description of the rest of the items in the most readable manner.

[0043] An example of a process performed by the output formatter 17 at step S25 in FIG. 8 will be described below. Assumed that the first presentation items of search results are "company name, location, representative, product class". All the items contained in the database 15 have not always values, but, for example, the "representative" item fields may be all blank in a data set of search results. In such a case, the output formatter 17 excludes item "representative" from the presentation items to avoid the presentation of useless items containing no information.

[0044] For example, if the following first presentation items are provided,

Company name	Representative	Location
Company A Japan	.....	Fuchu-machi, Toyama prefecture
Factory B	.....	Head quarters, Uozu, Toyama prefecture
Research institute C	.....	Joyo, Kyoto

then the "representative" item is removed because it contains no value. The final presentation items would be as follows:

Company name	Location
Company A Japan	Fuchu-machi, Toyama prefecture
Factory B	Head quarters, Uozu, Toyama prefecture
Research institute C	Joyo, Kyoto

[0045] As described above, if search results do not include an entry containing "representative" data and "representative" is selected as a presentation item, the item is excluded from the presentation items to eliminate redundancy, thereby enabling a readable presentation.

[0046] An example of an embodiment of the search criteria generator 13 will be described below with reference to FIG. 10. In this example, whether a search will succeed or not is determined before actually executing the search through the database 15, and if it is determined that it will fail, the user is informed of the cause of the failure. For this purpose, an item database containing all the values for the item that are held in the database 15 is provided for each of items in the database 15 that are to be searched through. In the example shown in FIG. 10, a location database (DB) 30, a product class DB 31, and a price DB 32 are provided.

[0047] The search criteria generator 13 first generates search criteria through the process described above with respect to FIG. 5 (step S30). Then it checks the criteria of each item as follows. An item specified as criteria is extracted from the generated search criteria (step S31). Then an item database (DBs 30 through 32) corresponding to the specified criteria item is searched for values specified as the criteria (step S32). Based on the result of the search, it is determined whether the value is contained in the item database, DBs 30 through 32, (step S33). If the value is found, the process proceeds to step S35. Otherwise, an alert message is output for informing the user that no entry is found in the database 15 that meets the criteria in the search request (step S34).

[0048] The above-described process is repeated for all the search items (step S35). Then, the search executor 14 searches through the database 15. If the alert message is provided to the user at step S35, the search executor 14 does not search the database 15, indicating a search failure.

[0049] A specific example of this will be described below. Suppose that a search request, "Want to know about PC makers in the Antarctic", for example, is provided. Then "SELECT company information WHERE product class = PC and location = Antarctic" as the search criteria are generated for this search request. The product class DB 31 is searched for an entry having a specified product class value, "PC" and the location DB 30 is searched for an entry having a specified location value of "Antarctic". As a result of the above-described item DB search, if the entry having the value of "Antarctic" is not found in the location DB 30, "No entry having the maker location "Antarctic was found " is provided as an alert message to the user.

[0050] Because it is indicated to the user that no data meeting the criteria "location = Antarctic" is found in the database 15 by the item database search as described above, the user can identify the cause of the search failure.

[0051] FIG. 11 shows an example of another embodiment of the search criteria generator 13. In this example, like the example shown in FIG. 10, an item DB containing all the values for the item that are held in the database 15 is provided for each of items in the database 15 that are to be searched through. If no entry that matches a value specified in a search request is found in an item DB, similar entries are retrieved and presented to the user as alternative item value candidates from which the user can make a selection.

[0052] The process for generating the first search criteria (step S40), extracting a criteria item from the search criteria (step S41), searching through an item DB (S42), and searching for the value (step S43) is the same as that performed at steps S30 through S33 in FIG. 10.

[0053] If an exact match is not found in the item DB through the item-by-item search, the item DB is searched for a value similar to the value (step S44). If a similar value is found in the item DB (step S45), the value is presented to the user to suggest replacing the value specified by the user as criteria with the presented value and prompt the user to confirm the suggestion. If a plurality of similar values are found, all of them are presented to the user so that the user can select one from them. If the user accepts the selection of a value, the search criteria are modified accordingly (step S46). Steps S41 through S46 are repeated for all criteria items in the search criteria generated at step S40 (step S47). Then, the search criteria are provided to a search executor 14 for executing the search.

[0054] A specific example of this will be described below. Suppose that a search request, "Want to know about instruments developed by マイクロハード (maikurohado)", is provided, for example. Then, a search command in which search criteria, "SELECT product information WHERE product class = instrument and company name = マイクロハード", is specified, is generated. An item DB (company name DB) is searched for an entry having the specified company name, "マイクロハード". If the entry is not found, the item DB is further searched for an entry having a value similar to the specified value, "マイクロハード".

[0055] The similar value may be provided by transliterating "マイクロハード" to a word, "Maikurohado", written with Roman letters. Then, the item DB is searched by using "Maikurohado" as a key. If "Microhard" is found as the most similar entry, the following alert message and a message for alternative value confirmation are provided to the user:

[0056] "No entry having the value "マイクロハード" was found."

[0057] "A similar value "Microhard" was found. Do you replace your search criteria with this value and continue the search?"

[0058] If the user accepts this alternative criteria, the search criteria in the search command are modified as follows:

[0059] "SELECT product information WHERE product class = instrument and company name = Microhard".

Then the search command is provided to the search executor 14.

[0060] As described above, if no entry matches the organization name, "マイクロハード", written in Katakana and the company name, "Microhard", which is written in Roman characters and has similar pronunciation, is found in the item DB, the modification of the search criteria is suggested to the user and search results that the user wants to obtain can be presented to the user.

[0061] FIG. 12 shows an example of another embodiment of the search criteria generator 13. In this example, the search criteria generator 13 presents to the user an alternative value to be specified for an item, and, if it is accepted by the user, the pair of an originally specified value and the alternative value are stored as synonymous words in an item and may be used in generating search criteria for the subsequent search requests to automatically widen the criteria.

[0062] First, the search criteria generator 13 generates search criteria (step S50). The process at step S50 is the same as that at steps S40 through S47 in FIG. 11. Then, the search criteria generator 13 determines whether a thesaurus is available or not (step S51) and, if the thesaurus is available, it is used to widen the search criteria (step S52).

[0063] It also determines whether the search criteria have been modified using an alternative criteria (at step S46 in FIG. 11) (step S53), and if modified, presents the alternative criteria to the user (step S54) to ask the user whether the user accepts the input of the criteria into the thesaurus. If the user accepts the input (step S55), the alternative criteria, that is, the pair of the originally specified value and the alternative value, are entered into the thesaurus (step S56).

[0064] A specific example of this will be described below. It is assumed that the search criteria, "Want to know about instruments developed by マイクロハード" in the above-described example are provided and the alternative criteria in which "マイクロハード" in the search criteria is replaced with "Microhard" are accepted by the user. At step S56 in FIG. 12, the pair of "マイクロハード" and "Microhard" are entered into a thesaurus. Then, if a search request, "What kind of disk drive is developed by マイクロハード?", are provided, a search command that "SELECT product information WHERE product class = disk drive and company name = マイクロハード" are specified as a search criteria is first generated. Then the thesaurus is referenced and the search criteria are widened as follows:

[0065] "SELECT product information WHERE product class = disk drive and company name = マイクロハード or Microhard" (step S52 in FIG. 12).

[0066] As described in this example, if the user accepts the replacement of "マイクロハード" with "Microhard", the pair of these words are entered into the thesaurus. When the user inputs "マイクロハード" in the subsequent processes, "Microhard" is searched for besides "マイクロハード" through widening criteria by using the thesaurus, thereby allowing search results desired by the user to be presented in the subsequent searches without an acceptance procedure by the user.

[0067] An example which the search executor 14 repeats the search mentioned below will be explained. The search criteria generator 13 generates the search criteria for the first search to be performed by a search executor 14. It is determined whether the search by a search executor 14 has succeeded or not, considering number of the search result. If it is determined that the search has failed, the search criteria are widened so as to increase the number of searched sets. Further the search is repeated by the search executor 14 until the search succeeds or further widening of the search criteria becomes impossible.

[0068] FIG. 13A and B illustrate the example of search criteria widening. As shown in FIG.13A, when search criteria are generated in the search criteria generator 13 (step S60), the output from the search criteria generator 13 is provided to the search executor 14, where the search is executed (step S61). It is determined whether the search has succeeded or not (step S62), and, if it is determined that the search has succeeded, the search process will end. If it is determined that the search has failed, a process for widening the search criteria is performed in a search criteria widening section (not shown) in the search criteria generator 13 (step S63), and the search is performed again by the search executor 14. The process continues until the search succeeds or further widening of the search criteria becomes impossible.

[0069] Suppose that a search request 20 is "Want to know about software for translating Japanese into Chinese" as shown in FIG. 13B. Then the search criteria generator 13 first generates a search command in which "SELECT product information WHERE product class = software and Japanese and Chinese and translation" of search criteria 22 are specified based on the result of an analysis by a search request analyzer 12 and sends it to the search executor 14. If the search fails because no data meeting the search criteria is found when the database 15 is searched for by the search executor 14, keywords are removed from the product class one by one, the leftmost first, to widen the search criteria. For example, "software" of the keyword is removed from the search criteria and the search is performed again by the search executor 14 based on "SELECT product information WHERE product class = Japanese and Chinese and translation" as a search criteria 22'. If the search succeeds, a query topic is extracted by a query intention analyzer 16, presentation items are selected by an output formatter 17, then the search results such as an output 26 are presented to the user.

[0070] The probability of success of a search increases because the search criteria are automatically widened by removing "software" of the first keyword, and the search is executed again if the search fails because the first criteria specified ("software and Japanese and Chinese and translation) is too narrow, as described above. Furthermore, if the search fails, the next keyword, "Japanese", is removed from the criteria similarly and entries that meet the criteria are searched for.

[0071] FIG. 14A,B and C show different examples of the method for widening search criteria. In an example shown in FIG. 14A, keywords in a search criteria item are removed one by one in order from the foremost (or the last) one like the example described with respect to FIG. 13B.

[0072] An example shown in FIG. 14B has a configuration in which a related-event DB 40 related to a particular item is provided and a particular item in a database to be searched is associated with a particular event. In this configuration, the related-event DB 40 is referenced and the value specific for a particular item is extended to a value specific for an item associated with an event relating to an event associated with that item. That is, if an item associated with an event is specified as search criteria and an event relating to that specified value exists, the name of the item is replaced with the related event.

[0073] Suppose that a search request states "Want to know about robots developed by Company A Japan." Then the search criteria generator 13 generates a search command in which the following search criteria are specified.

[0074] "SELECT product information WHERE product class = robot company name = Company A Japan event type = development"

[0075] If the search based on this search criteria fails, a related event, "sales", is retrieved from the related-event DB 40 and "development" in the search command is replaced with "sales" to widen the search criteria. As a result, the search is re-executed by using the following command.

[0076] "SELECT product information WHERE product class = robot company name = Company A Japan event type = sales"

[0077] Thus, in response to the search request, "Want to know about robots developed by Company A Japan", development events whose subject is Company A Japan are searched for, and if the search fails, the criteria widening operation is performed by using a related event to generate a search criteria that "development" of the event type is replaced with "sales" because the related-event DB 40 contains "sales" of the event relating to "development" of the original event. Therefore, robots sold by Company A Japan are searched for instead of robots developed by Company A Japan.

[0078] FIG. 14C shows an example of widening search criteria in which a database 15 to be searched is composed of tagged structured text such as XML. When the text-base database structured using tag information is searched, user's search request is eventually converted into value specific for each tag item.

[0079] In this embodiment, tag inclusion relation data 41 indicating inclusion relation of tags is provided beforehand. The tag inclusion relation data 41 may be data that <product information> has <sales information> as a superordinate conception, and <product class>, <product name>, and <price> as a subordinate conception, for example. In this example, if a value is specified for a tag and a specified tag has a superordinate tag that covers a broader range, the name of the tag is replaced with the superordinate tag.

[0080] Suppose that the following search criteria are generated initially.

[0081] "SELECT sales information WHERE company name = Company A Japan product name = medical software"

[0082] If a database 15 is searched based on this search criteria and the search fails, "product name" is replaced with "product information" to widen the search criteria, because "product information" which is the superordinate tag of "product name" covers a broader range including "product name".

[0083] "SELECT sales information WHERE company name = Company A Japan product information = medical software"

[0084] As mentioned above, a tag in the search criteria is replaced with its superordinate tag that covers a broader range to widen the search criteria. This criteria widening can address a case where a wrong tag is inserted in search criteria due to an error in tag analysis.

[0085] FIG. 15 shows an example of a process for prompting a user to review a search area when it is found in advance that a search will fail. First, in response to a search request input by the user, a search request analyzer 12 analyzes the search request (step S70). If the analysis fails due to an ambiguous expression in the search request (step S71), an alert message is provided to the user to prompt the user to change the expression (step S72). If the search request is analyzed successfully, then a search criteria generator 13 generates search criteria (step S73). If the generation of the search criteria fails due to the presence of an unknown word, for example (step S74), a notifying message is provided to the user to prompt the user to check to see if the search area is correct (step S75). If the search criteria are generated successfully, a search executor 14 performs the search (step S76).

[0086] Suppose that the search request states "Want to know about recent earthquakes in India". If the generation of search criteria fails because the analysis of the search request yields a result, "<time> recent </time> <unknown word> earthquakes in India </unknown word> <search request> want to know about </search request>", which contains the unknown word, "earthquakes in India", an alert message is output for prompting the user to check to see if the area of the query is correct. Here, an unknown word is a word that is not contained in the analysis dictionary 23 described earlier with respect to FIG. 4 or a word that does not have a corresponding word in a database 15.

[0087] FIG. 16 shows an example of a process for determining a search area, wherein a list of keywords unique to each of various areas is provided beforehand, the list is used to determine the area of a search request, and, if it is determined that the search request is in an area that is not addressed by a system, an alert is provided to the user to inform the user that the area is not addressed by the system.

[0088] A search request is input and analyzed in a search request analyzer 12 (step S80), and a search request phrase is removed (step S81). Then keywords are extracted (step S82) and keywords that are contained in each area of an area keyword vector are counted (step S83). The area keyword vector is pre-registered data indicating whether or not a keyword relates to any of various areas such as "politics", "economy" and "sport". For example, it indicates that keywords such as "baseball", "golf", and "player" belong to or closely relate to the "sport" area.

[0089] Keywords appearing in the search request are counted on an area basis and the area containing the most keywords is regarded as the search area (step S84). It is determined whether the search area matches an area in a database 15 that is to be searched (step S85) and, if it does not match any of the areas in the database, an notifying message is provided to the user to indicate that there is no matching area (step

S86). If it matches an area in the database, a search criteria generator 13 generates search criteria (step S87).

[0090] FIG. 17 shows an example of another embodiment of the search criteria generator 13. In this example, if an unknown word topic is found in a search request, search criteria for items on a main item list which is provided in advance are generated for that unknown topic, a search is performed in each item, and the results are presented to the user.

[0091] First, a core topic of a query is selected from the results of the analysis of the search request (step S90). Then, it is determined whether the topic is an unknown word (step S91). If it is an unknown word, it is replaced with a main category (step S92). A main category is an item placed on the main item list in advance. Typically, a plurality of items are included in the main item list. Then, search criteria are generated in which any unknown word is resolved (step S93).

[0092] For example, suppose that the search request is "Want to know about an access" and a search request analyzer 12 outputs "<unknown word> access </unknown word> <search request> want to know about </search request>". Because the <access> is an unknown word, it is unknown under which item the "access" falls as a specified value. If the main item list includes "company name" and "product class", category for <access> are assumed one by one to generate search criteria as follows.

[0093] • When "access" is assumed to be an organization, search criteria, "SELECT company information WHERE company name = access", is generated, and

[0094] • when "access" is assumed to be a product, search criteria, "SELECT product information WHERE product class = access", is generated.

[0095] As described above, if a search criteria generator 13 fails to generate search criteria for the search request that is "Want to know about an access", under items in a database, the search criteria generator 13 generates search criteria in which a criterion part is assumed to fall under a main item contained in the database. That is, if "organization" and "product" as main items are contained in the database, two searches are performed: one in which "access" is assumed to be a company name and one in which "access" is assumed to be a product name. Then the results are presented to the user.

[0096] FIG. 18 shows an example of another embodiment of the search criteria generator 13 and search executor 14. In this example, a database 15 to be searched is a text base structured by using tags. If the analysis of a search request yields a query about a word for which no tag is specified, the word is first used to perform a simple keyword search without tags, then the results of the search are classified by tag applied to words and presented to the user.

[0097] First, a search criteria generator 13 generates search criteria (step S100). If a tagged word is inquired about and the analysis of the correspondence between an item and the specified value fails and therefore no search criteria can be generated (step S101), a keyword is first extracted from the search request and the extracted keyword is used to perform a simple search through a database 15 without a tag (step S102). A tag near the keyword used, that is, a tag covering a small area including the keyword, is extracted from the results of the search (step S103), then the search results are classified by tag and presented to the user (step S104).

[0098] An example of a text complement process during the entry of text into a database 15 will be described below with respect to FIG. 19. It is assumed that the database 15 to be searched is composed of text data structured using tags, such as XML. Secondary databases (DBs) are provided for essential tags. A correspondence table 50 indicating correspondences between tags and the secondary databases is also provided as shown in FIG. 19A. In this example, a secondary database DB2 as shown in FIG. 19B is provided for a tag of "representative" and another secondary database DB3 is provided for a tag of "location."

[0099] Also provided is an essential item list 51 containing essential items relating to subjects covered by the database 15 to be searched for performing a process shown in FIG. 20 during the entry of text into the database 15.

[0100] First, an essential item is extracted from the essential item list 51 (step S110). Then, it is determined whether the essential item is contained in text to be entered in the database 15 (step S111). If the text data to be entered concerns "organization", then items "organization name", "location", and "representative" would be essential items. On the other hand, if it concerns "product", items "product class", "product name", and "price" would be essential items. If the text contains organization name information and does not contain representative and location information, the correspondence table 50 between tags and secondary DBs is referenced to determine whether there are secondary databases for representatives and locations (step S112). In the example shown in FIG. 19, there are secondary databases DB2, DB3 for them, therefore secondary databases DB2, DB3 are searched for a representative and location associated with an organization name and a location contained in the text, respectively (step S113), and retrieved values are inserted into the original text to be entered in the database (step S114).

[0101] Thus, if the text to be entered is "<organization name> Factory A </organization name>" as shown in text 52 in FIG. 19D, Complementing the data is executed by using the secondary databases DB2, DB3 and "<organization name> Factory A </organization name> <representative> President Yamazaki </representative> <location> Kawasaki </location>" as shown in text 53 in FIG. 19E is actually entered in the database 15.

[0102] The above-described process is repeated for all essential items on the essential item list 51 (step S115).

[0103] Fig.20 shows a process to complete data during the entry of the text data in a database. However missing data may be added during a search process by using secondary databases . FIG. 21 shows an example of this.

[0104] First, if it is detected that the description of a value in a search item is not contained in a database 15 (step S120), the secondary databases is searched for a key item by using a value in the database 15 (step S121). The process will be described by using an example in which secondary databases similar to those in the example in FIG. 19 are provided. If the database 15 contains data, "Factory A", and does not contain information about its representative and location, the information about the representative and location is retrieved from secondary databases. This information is used to determine whether a retrieved value meets criteria (step S122). If it meets the criteria, the item criteria are considered to have succeeded. Otherwise, the item criteria are considered to have failed.

[0105] For example, in response to a search request that is "Find companies selling printers located in Kawasaki", a search criteria generator 13 generates the following search criteria:

[0106] "SELECT company information WHERE location = Kawasaki and product class = printer".

[0107] Then it is determined whether there is a secondary database for each item in the search criteria. If a secondary database DB3 as shown in FIG. 19B is found for "location", the location criterion in the search criteria is used to search the secondary database DB3 and search results such as "Factory A", "Company C",… are obtained from the database 15. The results are checked to see if they meet the other criteria (product class = printer") and obtain "Factory A". Search sets are summed to obtain final search results.

[0108] If a database to be searched is a text base structured with tags, values for individual items may be extracted and first entered in individual databases during the entry of text into the text database, then a group of spellings resembling each other may be extracted from the individual databases after the completion of the entry, thereby enabling a precise detection of variations in notation compared with that in a case where the entire text is searched.

[0109] As described above, according to the present invention, the results of a search through a database performed in response to user's request are reduced to an appropriate quantity, formatted according to user's requirements, and provided in organized, readable form.